Airfoil/Flap Example Case*

The NPARC Alliance
NASA Glenn Research Center
Cleveland, Ohio
USAF Arnold Engineering Development Center
Tullahoma, Tennessee

*This is an unnumbered version of this document, created September 24, 2003. Please send corrections, additions, ideas, etc., to Charlie Towne at *towne@grc.nasa.gov* or Mark Fisher at *mark.s.fisher@boeing.com*.

Contents

1 Introduction

This document describes the use of Wind-US to solve a problem using a Chimera, or overset, grid. It is primarily intended to demonstrate the use of GMAN for Chimera grids.[1]

The geometry is a 2-D airfoil with a flap, shown in Figure 1.

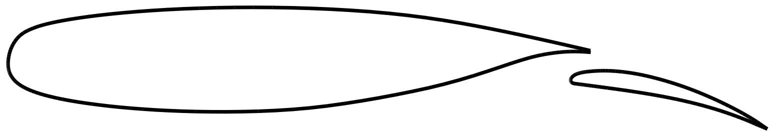


Figure 1: Airfoil/flap geometry.

Separate C grids have been created for the airfoil and the flap, with the flap grid overlaying the airfoil grid, as shown in Figure 2.

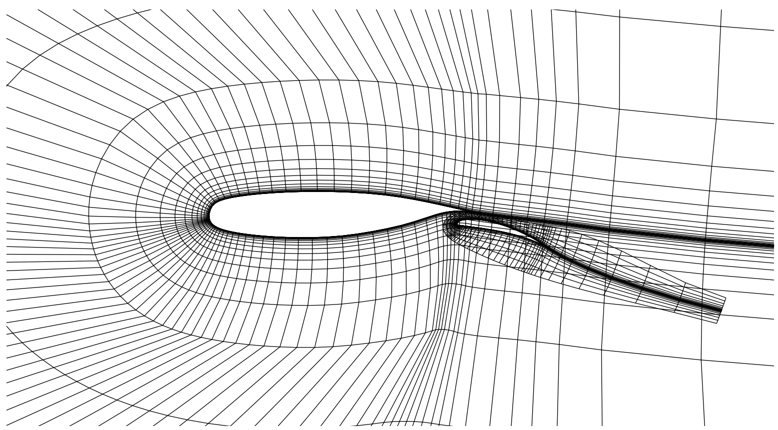


Figure 2: Original airfoil/flap grids.

The outer boundary of the airfoil grid actually extends far beyond the airfoil, but is expanded in Figure 2 to show the details near the surface. The two grids are contained in a multi-zone formatted PLOT3D xyz file. The airfoil grid is zone 1, with 205×59 points, and the flap grid is zone 2, with 179×33 points. In the figure, every other point in both directions, for both grids, has been omitted.

In the sections that follow:

- The *cfcnvt* utility is used to create a common grid (*.cgd*) file from the existing PLOT3D file.
- GMAN is used to cut a hole in the airfoil grid around the flap, define the fringe boundaries for both grids, and set the remaining boundary conditions. This discussion makes up the bulk of the document.
- A Wind-US input data (*.dat*) file is created, and Wind-US is run to solve for the flow field.
- The resulting common flow (*.cfl*) file is used with CFPOST to create PLOT3D xyz and q files, and the results are displayed.

[1]The various files discussed in this document may be downloaded from the Wind-US documentation WWW site, at `http://www.lerc.nasa.gov/www/winddocs/gman/example/downloading.html`.

2 Creating the *.cgd* File

The PLOT3D xyz file containing the overlayed grids is named *demo.grid*, and is a 3D (although with $k_max = 1$), multi-zone, formatted, single-precision file without an IBLANK array. A *.cgd* file named *demo.cgd* was created using *cfcnvt*, as shown in the following runstream. Lines in slanted type were typed by the user.

```
cfcnvt
          ******* Common File Convert Utilities ******
 CFCNVT - Version 1.29 (last changed 1997/10/14 16:05:43)

   0: Exit program
   1: MDA/ZDA/FDA/PDA conversions
   2: Import   a Common File
   3: Compress a Common File
   4: Break Common File into multiple transfer files
   5: Combine multiple transfer files into Common File
   6: Append one Common File to another
   7: Convert Common File binary to a text file
   8: Convert Common File text to a binary file
   9: Convert ascii ZDF to Common File ZDF
  10: Convert Common File ZDF to ascii ZDF
  11: Convert PLOT3D   file to Common File
  12: Convert GASP     file to Common File
  13: Convert OVERFLOW file to Common File
  14: Convert Common File to OVERFLOW file
  15: Convert CFPOST GPU file to Common File GPC

  Enter the number from one of the above requests
11

   PLOT3D file type menu

   0: Main menu
   1: Convert a PLOT3D Grid     (.x) file to CFS.
   2: Convert a PLOT3D Solution (.q) file to CFS.

  Enter the number from one of the above requests
1

   PLOT3D Number of Grids menu

   0: Main menu
   1: PLOT3D Single zone format.
   2: PLOT3D Multi  zone format.

  Enter the number from one of the above requests
2

   PLOT3D Zone dimension menu
```

```
   0: Main menu
   1: PLOT3D 2d zone format.
   2: PLOT3D 3d zone format.

  Enter the number from one of the above requests
2

   PLOT3D Format menu

   0: Main menu
   1: PLOT3D Formatted   (ASCII).
   2: PLOT3D Unformatted (sequential binary).

  Enter the number from one of the above requests
1

   PLOT3D Iblank menu

   0: Main menu
   1: PLOT3D grid with    IBLANK format.
   2: PLOT3D grid without IBLANK format.

  Enter the number from one of the above requests
2

   PLOT3D Precision menu

   0: Main menu
   1: PLOT3D Single precision format.
   2: PLOT3D Double precision format.

  Enter the number from one of the above requests
1
  Enter PLOT3D .x file to convert with suffix
demo.grid
  Enter output Common File name with suffix
demo.cgd
  Global maximums set as follows:
  mimax         205
  mjmax          59
  mkmax           1
  mpts:       12095
  recl:         563
 Processing zone ZONE   1
 Writing mesh data
 Processing zone ZONE   2
 Writing mesh data

   0: Exit program
   1: MDA/ZDA/FDA/PDA conversions
   2: Import   a Common File
```

```
  3: Compress a Common File
  4: Break Common File into multiple transfer files
  5: Combine multiple transfer files into Common File
  6: Append one Common File to another
  7: Convert Common File binary to a text file
  8: Convert Common File text to a binary file
  9: Convert ascii ZDF to Common File ZDF
 10: Convert Common File ZDF to ascii ZDF
 11: Convert PLOT3D   file to Common File
 12: Convert GASP     file to Common File
 13: Convert OVERFLOW file to Common File
 14: Convert Common File to OVERFLOW file
 15: Convert CFPOST GPU file to Common File GPC

 Enter the number from one of the above requests
0
```

3 Using GMAN

GMAN was then used in graphics mode to cut a hole in the airfoil grid around the flap, define the fringe boundaries for both grids, and set the remaining boundary conditions. The file *demo.jou* is the journal file created during the interactive GMAN session, with the addition of comments describing the procedure.

The first step, obviously, is to start GMAN.

gman

```
                         ***** gman *****

           Select the desired version from the following list.

0) END
1) gman optimized version
2) gman_pre optimized version

Enter number of executable.......[1]:

            GMANPRE - Version 6.100 (last changed 1997/08/07 20:47:48)
                        Creating journal file 'gman.jou'.

  Enter SWITCH or GRAPHICS to change to graphics mode.

 GMAN:
```

At this point, you could type `@demo.jou` to execute the commands stored in the journal file *demo.jou*. You could also enter commands individually at the `GMAN:` prompt. Or, you could enter `SWITCH` or `GRAPHICS` to enter graphics mode.

The following sections describe in detail the use of GMAN for the airfoil/flap configuration. The graphics mode steps are on the left, with the Main Menu steps left-aligned and the Menu Options indented. Most of these are accomplished in GMAN by clicking on the listed menu item using the left mouse button. A few require entering text in the prompt area at the bottom of the screen. (See the "Graphical User Interface Basics" section of the *GMAN User's Guide* for a description of the various sections of the GMAN screen layout.)

The command line equivalents are shown on the right. Note that, in general, several graphics mode steps become consolidated into a single command. A few of the commands are too long to fit on a single line, and are thus shown with the continued part indented. When using GMAN in command-line mode, however, they should be entered on a single line.

3.1 Opening the File

We first need to tell GMAN the name of the file containing the grid.

Graphics Mode	**Command Line Mode**
`Common File` *enter* `demo.cgd`	`FILE demo.cgd`

3.2 Cutting the Hole

Next, we need to cut a hole in the airfoil grid surrounding the flap.

The first step is to select the zone containing the airfoil grid, and to select the "Overlapping" boundary type. (This boundary type is useful for setting boundary conditions on interior grid points, as well as for Chimera grids.)

Graphics Mode	Command Line Mode
`BOUNDARY COND.`	`ZONE 1`
`1:` (*from Zone List*)	
`OLAP`	`BOUNDARY OLAP`

We need to tell GMAN the procedure we're going to use to define the hole. In this case the hole will be defined by grid surfaces that form a closed region around the flap.

Graphics Mode	Command Line Mode
`MODIFY BNDY`	`CUTTER SURFACE TYPE`
`SHOW OTHER ZONE`	
`2:` (*from Zone List*)	
`PICK K-PLANE`[2]	
`OLAP GENERATION`	
`GENERATE HOLES`	
`SET/SHO CUTTER`	
`CUTTER SURFACE`	

Next, we select the surfaces from the flap grid that specify a closed region around the flap defining the hole.

The $j = 1$ surface in the flap grid (i.e., the flap surface itself) could be used. However, the flap grid is tightly packed near the flap surface to resolve the boundary layer, while the airfoil grid spacing in this region is much larger. Ideally, the exchange of information between the two grids should occur in regions where the grid cells are similar in size.

Using a grid line farther out, near the outer edge of the flap zone, would result in a hole bigger than the flap grid. Since the boundary conditions around the hole for the solution in the airfoil zone will be computed from the solution in the flap zone, this would be unacceptable.

For these reasons, the $j = 26$ grid line around the flap is used.

[2] At this point, you may want to adjust the view in the window by using the middle and right mouse buttons, or the `Center` choice in the Display Options menu, to enlarge the the view around the flap.

Graphics Mode	Command Line Mode
`SET SRF ZON`	`CUTTER SURFACE IN ZONE 2 I20 J26 K1`
enter `2`	`TO I160 J26 K1 INCREMENT 1`
`SET SURF RANGE`	
enter `20 26 1`	
enter `160 26 1`	
`SET SURF RANGE`	`CUTTER SURFACE IN ZONE 2 I20 J1 K1`
enter `20 1 1`	`TO I20 J26 K1 INCREMENT 1`
enter `20 26 1`	
`SET SURF RANGE`	`CUTTER SURFACE IN ZONE 2 I160 J1 K1`
enter `160 1 1`	`TO I160 J26 K1 INCREMENT 1`
enter `160 26 1`	

The surfaces defining the hole cutter are shown in Figure 3, along with both grids in the region of the flap.

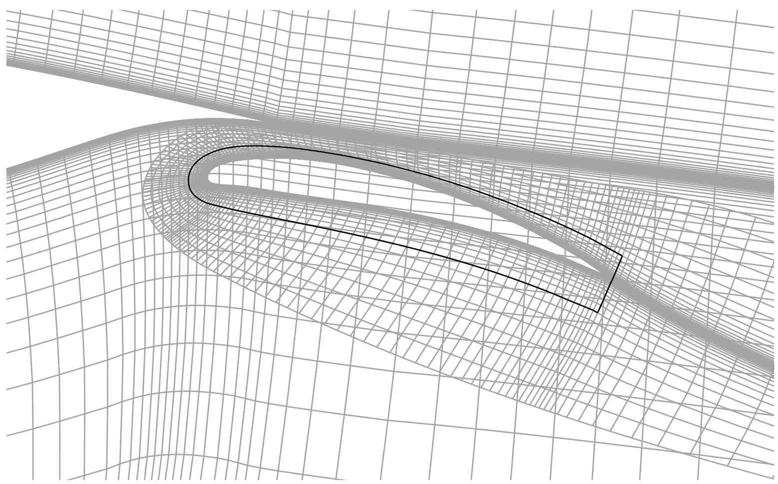


Figure 3: Surfaces defining hole cutter.

All overlapping boundaries and holes must have a unique label, greater than six. The `Label` menu choice is in the Subregion Selection area in the lower-right corner of the GMAN window.

Graphics Mode	Command Line Mode
`Label`	`LABEL 7`
enter `7`	

Finally, we can create the hole.

Graphics Mode	Command Line Mode
`GENERATE HOLE`	`GENERATE HOLE`

The resulting airfoil grid, with the hole cut into it, is shown in the region of the flap in Figure 4.

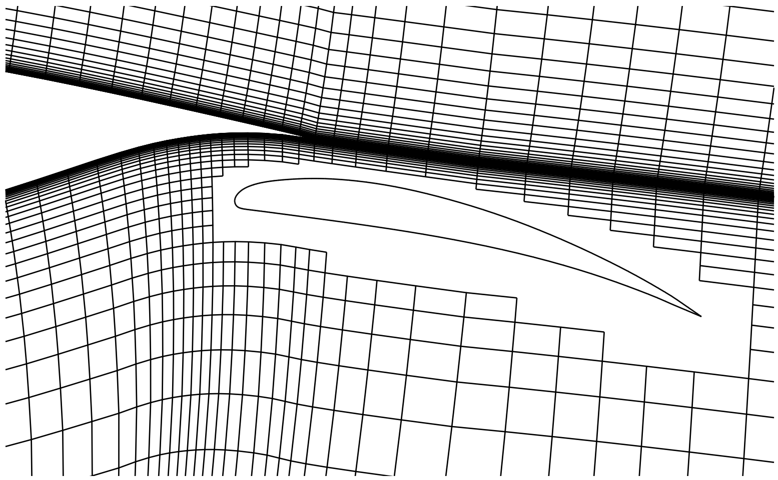


Figure 4: Airfoil grid with hole.

3.3 Defining the Fringes

The next step is to define the fringes and set boundary conditions on them. For the airfoil grid, the fringes will be defined as the hole boundary.

<u>**Graphics Mode**</u>	<u>**Command Line Mode**</u>
`OLAP GENERATION`	
`GENERATE FRINGE`	
`   FRG MODE HOLE`	`FRINGE-MODE HOLE`
`GENERATE FRINGE`[3]	`GENERATE FRINGE`
`OLAP GENERATION`	
`SET FRINGE BND`	`SET FRINGE BOUNDARY COUPLED TO`
`COUPLE`	`   ZONE 2 BOUNDARY OLAP`
`   2:` (*from Zone List*)	
`   OLAP`	
`COUPLE`	
`BOUNDARY COND.`	`UPDATE`
`   YES, SAVE CHANGES`	

The flap grid is completely embedded in the airfoil grid, and the outer boundaries do not overlap any solid boundaries. Cutting a hole is thus not necessary, and we can use the outer boundaries as the fringe.

[3] From the Main Menu, not the Menu Trail

Graphics Mode	Command Line Mode
PICK ZONE/BNDY	ZONE 2
2: (*from Zone List*)	BOUNDARY OLAP
OLAP	
MODIFY BNDY	
GENERATE FRINGE	
FRG MODE OUTER	
BOUND I1 ON	FRINGE-MODE OUTER BOUNDARY ON I1
BOUND IMAX ON	FRINGE-MODE OUTER BOUNDARY ON IMAX
BOUND JMAX ON	FRINGE-MODE OUTER BOUNDARY ON JMAX
Label	LABEL 8
enter 8	
GENERATE FRINGE[4]	GENERATE FRINGE
OLAP GENERATION	
SET FRINGE BND	SET FRINGE BOUNDARY COUPLED TO
COUPLE	ZONE 1 BOUNDARY OLAP
1: (*from Zone List*)	
OLAP	
COUPLE	
BOUNDARY COND.	UPDATE
YES, SAVE CHANGES	

3.4 Setting the Remaining Boundary Conditions

The remaining boundary conditions that need to be set for the C grid around the airfoil are the downstream outflow boundaries, the outer boundary, the C grid cut line, and the airfoil surface. The downstream and outer boundaries of the flap grid have already been coupled to the airfoil grid, so all that remain are the C grid cut line and the flap surface.

The downstream outflow boundaries for the airfoil grid are at `I1` and `IMAX`, and are defined as outflow boundaries.

Graphics Mode	Command Line Mode
PICK ZONE/BNDY	ZONE 1
1: (*from Zone List*)	
I1	BOUNDARY I1
MODIFY BNDY	OUTFLOW
CHANGE ALL	
OUTFLOW	
BOUNDARY COND.	UPDATE
YES - UPDATE FILE	
PICK ZONE/BNDY	
IMAX	BOUNDARY IMAX
MODIFY BNDY	OUTFLOW
CHANGE ALL	
OUTFLOW	
BOUNDARY COND.	UPDATE
YES - UPDATE FILE	

[4] From the Main Menu, not the Menu Trail

The outer boundary of the airfoil grid is at JMAX, and is defined as a freestream boundary.

Graphics Mode	Command Line Mode
PICK ZONE/BNDY	
JMAX	BOUNDARY JMAX
MODIFY BNDY	FREESTREAM
CHANGE ALL	
FREESTREAM	
BOUNDARY COND.	UPDATE
YES - UPDATE FILE	

The C grid cut line and the airfoil surface are both at J1. First, we'll couple the grid to itself, using "point match" coupling. This will define the cut line as a coupled boundary, leaving the points on the airfoil surface undefined.

Graphics Mode	Command Line Mode
PICK ZONE/BNDY	
J1	BOUNDARY J1
MODIFY BNDY	CPMODE POINT MATCH
COUPLE	COUPLED TO ZONE 1 BOUNDARY J1
1: (*from Zone List*)	
SET COUP MODE	
POINT MTCH ON	
SEL OTHER BNDY	
J1:	
COUPLE	

Now, we'll set a viscous wall condition at the remaining undefined points, and write the changes to the *.cgd* file.

Graphics Mode	Command Line Mode
CHANGE BY TYPE	UNDEFINED IS VISCOUS WALL
UNDEFINED	
VISCOUS WALL	
BOUNDARY COND.	UPDATE
YES - UPDATE FILE	

Like the airfoil grid, the cut line and the flap surface in the flap grid are both at J1. The boundary conditions there are set in the same way as in the airfoil grid.

Graphics Mode	**Command Line Mode**
`PICK ZONE/BNDY`	
`2:` (*from Zone List*)	`ZONE 2`
`J1`	`BOUNDARY J1`
`MODIFY BNDY`	`COUPLED TO ZONE 2 BOUNDARY J1`
`COUPLE`	
`2:` (*from Zone List*)	
`SEL OTHER BNDY`	
`J1:`	
`COUPLE`	
`CHANGE BY TYPE`	`UNDEFINED IS VISCOUS WALL`
`UNDEFINED`	
`VISCOUS WALL`	
`BOUNDARY COND.`	`UPDATE`
`YES - UPDATE FILE`	

Finally, it's a good idea to check the boundary conditions to make sure all is OK.[5]

Graphics Mode	**Command Line Mode**
`TOP`	
`CHECK`	`ZONE ALL`
`CHECK BOUNDARY`	`CHECK BOUNDARY`
`PICK ZONE`	
`ALL` (*from Zone List*)	
`RUN BNDY CHKS`	

After hitting the `Enter` key to return to graphics mode, we can quit GMAN.

Graphics Mode	**Command Line Mode**
`END`	`EXIT`
`YES - TERMINATE`	

[5]This checks regular (non-fringe) boundaries. Fringe boundaries can also be checked, using `RUN FRNG CHKS` in graphics mode, or `CHECK FRINGE` in command line mode. Doing the fringe boundary check for this case yields several messages about volume ratios exceeding the tolerance. This is intended as a warning about differences in the sizes of the grid cells in the two zones in the area of the fringes. As noted earlier, the grid cells in this area should be similar in size. For this case, however, this does not seem to present a problem in the solution.

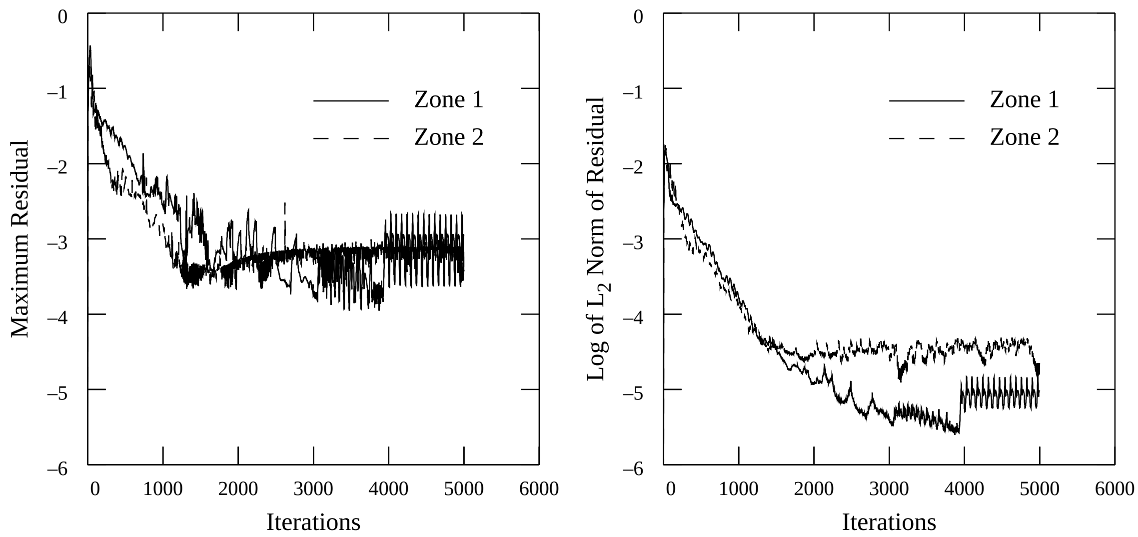
4 Running Wind-US

Listed below is the input data (*.dat*) file *demo.dat* used for this case.

```
Demo of airfoil/flap Chimera grid
Mach 0.6
First run

/ Inlet conditions
Freestream static 0.6 14.7 520. 0. 0.

/ Boundary conditions
Downstream pressure 14.7 zone 1

/ Viscous terms
Turbulence sst

/ Numerics
Cycles 500

End
```

Two runs were made for a total of 1000 cycles (5000 iterations) On an SGI Octane R10000 system, in serial mode, each run required just over 3200 seconds of CPU time. Genplot files for the maximum and L_2 residuals were generated using *resplt*, as described in the "Tutorial" section of the *Wind-US User's Guide*.

The residuals are shown in Figure 5(a) and Figure 5(b).



(a) Maximum residual

(b) L_2 norm of the residual

Figure 5: Computed residuals.

5 Results

Computed flow field results may be listed and/or plotted using CFPOST, as described in the *CFPOST User's Guide*. CFPOST may also be used to create PLOT3D xyz and q files for use as input to other post-processors, as described under "Examine the Results" in the "Tutorial" section of the *Wind-US User's Guide*.

The computed results are shown in Figure 6 in the form of constant static pressure contours.

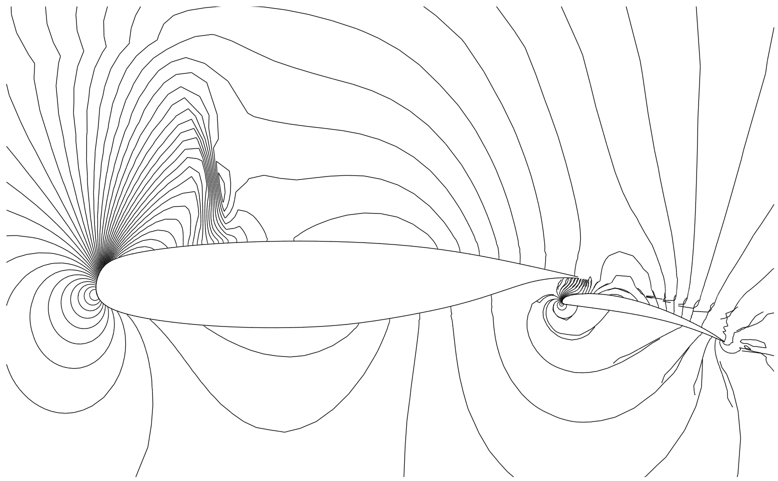


Figure 6: Computed static pressure contours.